Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (currently amended) A computer readable data storage medium storing software for supporting a plurality of intelligent messaging network servers in an intelligent messaging network, the software comprising:

a first code segment handling registration of a first intelligent messaging network server of said plurality of intelligent messaging network servers in said intelligent messaging network, wherein registration comprises storing a server id and a server type for said first intelligent messaging network server in a database storing server ids and server types for said plurality of intelligent messaging network servers;

a second code segment for connecting said first intelligent messaging network server to a second intelligent messaging network server of said plurality of intelligent messaging network servers; and

a third code segment encapsulating communication between said first intelligent messaging network server and said second intelligent messaging network server;

wherein a transport protocol used within said intelligent messaging network provides for networking services comprising at least one of message segmentation and reassembly, message retries, message duplication detection, and message ACK and NACK service, said networking services being performed without relying on either a client application and server application.

2.   (previously presented) The computer readable data storage medium of claim 1, wherein:

said first code segment specifies a server class for said first intelligent messaging network server.

3.   (previously presented) The computer readable data storage medium of claim 1, wherein:

said first code segment further specifies at least one of a packet header, an IP address, and a listener port.

4.   (previously presented) The computer readable data storage medium of claim 1, wherein:

said third code segment generates a standard packet for communication between said first intelligent messaging network server and said second intelligent messaging network server.

5.   (original) The computer readable data storage medium of claim 4, wherein the standard packet includes at least one of:

a header length;

protocol flags;

packet length;

database ID;

link station ID;

message ID;

customer ID;

port number;

network header; and

message body.

6.   (original) The computer readable data storage medium of claim 5, wherein the network header includes at least one of:

a compression indicator;

a security indicator;

a service type indicator,

a message type indicator; and

a server ID.

7.   (original) The computer readable data storage medium of claim 1, further comprising:

a fourth code segment encapsulating a transport header;

a fifth code segment notifying a sender of a success or failure of a transmission;

a sixth code segment segmenting messages over a pre-determined length into message segments;

a seventh code segment assembling messages segments into messages;

an eighth code segment resending messages that are not acknowledged within a pre-determined time;

a ninth code segment pacing a transmission of messages larger than a predetermined number of segments;

a tenth code segment detecting duplicate message segments; and

an eleventh code segment detecting duplicate messages.

8.   (original) The computer readable data storage medium of claim 1, further comprising:

a fourth code segment generating acknowledgement messages;

a fifth code segment processing the acknowledgement messages;

a sixth code segment compressing and decompressing messages; and

a seventh code segment encrypting and decrypting messages.

9.   (original) The computer readable data storage medium of claim 7, further comprising:

a twelfth code segment encapsulating a communication layer.

10.   (previously presented) The computer readable data storage medium of claim 8, further comprising:

an eighth code segment processing application specific messages; and

a ninth code segment providing special compression services; and

a tenth code segment providing special security services.

11.   (withdrawn) A computer readable storage medium comprising a software class providing graphical user interfaces, the class comprising:

a first code segment providing a base registry key for storage of server settings and an user interface for viewing or editing the server settings;

a second code segment providing screen based error logging; and

a third code segment for writing system errors to an event log.

12.   (withdrawn) The computer readable data storage medium of claim 11, wherein the third code segment specifies a batch file to be executed when a specified error occurs.

13.   (withdrawn) The computer readable data storage medium of claim 11, further comprising:

a fourth code segment providing a user interface for transport settings;

a fifth code segment logging each inbound and outbound message; and a sixth code segment displaying pre-selected statistics.

14.　(withdrawn) The computer readable data storage medium of claim 13, further comprising a seventh code segment providing a separate logging interface for logging application errors.

15.　(withdrawn) The computer readable data storage medium of claim 11, wherein the transport settings include at least one of: a maximum number of retries, a retry timeout interval, and a segment size.

16.　(withdrawn) The computer readable data storage medium of claim 11 wherein the pre-selected statistics include at least one of a number of messages sent/received and a number of ACK/NACK sent/received.

17.　(withdrawn) The computer readable data storage medium of claim 11, further comprising:

a fourth code segment providing a GUI for displaying a log of inbound/outbound messages;

a fifth code segment logging each inbound and outbound message; and a sixth code segment displaying pre-selected statistics.

18.　(withdrawn) A software development system for developing client applications, comprising:

a utility library;

a security library; and

a transport library, wherein the transport library is independent from both the security library and the utility library.

19.   (withdrawn) The system of claim 18, wherein the utility library comprises:

a first code segment for handling streaming input/output messages to the client application;

a second code segment providing compression services for messages;

a third code segment creating a message header for messages generated by the client application; and

a fourth code segment building authentication messages.

20.   (withdrawn) The system of claim 19, wherein the second code segment determines if a message is to be encoded and provides encoding services.

21.   (withdrawn) The system of claim 19, wherein the fourth code segment determines an authentication status.

22.   (withdrawn) The system of claim 18, wherein the transport library comprises:

a first code segment for specifying a target of a message generated by the client application;

a second code segment notifying a sender of a success or failure of a transmission;

a third code segment segmenting messages over a pre-determined length into message segments;

a fourth code segment assembling messages segments into messages;

an fifth code segment for resending messages that are not acknowledged within a pre-determined time;

a sixth code segment handling duplicate message segments; and

an seventh code segment handling duplicate messages.

23. (withdrawn) The system of claim 18, wherein the security library comprises:

a first code segment establishing a secret key between the client application and a server;

a second code segment for encrypting messages; and a third code segment for decrypting messages.

24. (currently amended) A method for supporting a plurality of intelligent messaging network servers in an intelligent messaging network, comprising:

providing registration of a first intelligent messaging network server of said plurality of intelligent messaging network servers in said intelligent messaging network, wherein registration comprises storing a server id and server type for said first intelligent messaging network server in a database storing server ids and server types for said plurality of intelligent messaging network servers;

providing connectivity of said first intelligent messaging network server and a second intelligent messaging network server of said plurality of intelligent messaging network servers; and

encapsulating communication between said first intelligent messaging network server and said second intelligent messaging network server;

wherein a transport protocol used within said intelligent messaging network provides for networking services comprising at least one of message segmentation and reassembly, message retries, message duplication detection, and message ACK and NACK service, said networking services being performed without relying on either a client application and server application.

25. (previously presented) The method of claim 24, further comprising:

specifying a server class for said first intelligent messaging network server during registration.

26. (previously presented) The method of claim 24, further comprising:

specifying at least one of a packet header, an IP address and a listener port during registration.

27. (previously presented) The method of claim 24, further comprising:

generating a standard packet for communication between said first intelligent messaging network server and said second intelligent messaging network server during encapsulation.

28. (previously presented) The method of claim 27, wherein the standard packet includes at least one of:

a header length;

protocol flags;

packet length;

database ID;

link station ID;

message ID;

customer ID;

port number;

network header; and

message body.

29.   (previously presented) The method of claim 28, wherein the network header includes at least one of:

a compression indicator;

a security indicator;

a service type indicator;

a message type indicator; and

a server ID.

30.   (previously presented) The method of claim 24, further comprising:

encapsulating a transport header;

notifying a sender of a success or failure of a transmission;

segmenting messages over a pre-determined length into message segments;

assembling the messages segments into messages;

resending messages that are not acknowledged within a pre-determined time;

pacing a transmission of messages larger than a pre-determined number of segments;

detecting duplicate message segments; and

detecting duplicate messages.

31.   (previously presented) The method of claim 24, further comprising:

generating acknowledgement messages;

processing the acknowledgement messages;

compressing and decompressing messages; and

encrypting and decrypting messages.

32.   (previously presented) The method of claim 30, further comprising:

encapsulating a communication layer.

33.   (previously presented) The method of claim 31, further comprising:

processing application specific messages;

providing special compression services; and

providing special security services.

34.   (withdrawn) A computer readable storage medium comprising a software class providing graphical user interfaces, the class comprising:

providing a base registry key for storage of server settings and an user interface for viewing or editing the server settings; and

providing screen based error logging; and writing system errors to an event log.

35.   (withdrawn) The computer readable data storage medium of claim 34, wherein the writing step specifies a batch file to be executed when a specified error occurs.

36.   (withdrawn) The computer readable data storage medium of claim 34, further comprising:

providing a user interface for transport settings;

logging each inbound and outbound message; and

displaying pre-selected statistics.

37.   (withdrawn) The computer readable data storage medium of claim 36, further comprising:

providing a separate logging interface for logging application errors.

38. (withdrawn) The computer readable data storage medium of claim 34, wherein the transport settings include at least one of:

a maximum number of retries, a retry timeout interval, and a segment size.

39. (withdrawn) The computer readable data storage medium of claim 34 wherein the pre-selected statistics include at least one of a number of messages sent/received and a number of ACK/NACK sent/received.

40. (withdrawn) The computer readable data storage medium of claim 34, further comprising:

providing a GUI for displaying a log of inbound/outbound messages; logging each inbound and outbound message; and

displaying pre-selected statistics.

41. (withdrawn) A method for developing client applications, comprising:

providing utility components for the client application to run; providing security components for encryption; and

providing transport components for data communication over wireless networks, wherein the transport components are independent from both the security components and the utility components.

42. (withdrawn) The system of claim 41, wherein the utility components include elements:

handling streaming input/output messages to the client application;

providing compression services for messages;

creating a message header for messages generated by the client application; and

building authentication messages.

43.   (withdrawn) The system of claim 42, wherein the compression services determine if a message is to be encoded and provides encoding services.

44.   (withdrawn) The system of claim 42, further comprising determining an authentication status.

45.   (withdrawn) The system of claim 41, wherein the transport components comprise elements: specifying a target of a message generated by the client application;

notifying a sender of a success or failure of a transmission;

segmenting messages over a pre-determined length into message segments;

assembling message segments into messages;

resending messages that are not acknowledged within a pre-determined time;

handling duplicate message segments, and

handling duplicate messages.

46.   (withdrawn) The system of claim 41, wherein the security components comprise elements:

establishing a secret key between the client application and a server;

encrypting messages; and

decrypting messages.

47. (currently amended) A software development kit (SDK), comprising:

registration components for handling registration of intelligent messaging network servers with an intelligent messaging network, wherein registration comprises storing server ids and server types for said intelligent messaging network servers in a database;

connectivity components for connecting said intelligent messaging network servers to one another; and

communication components for encapsulating communication between said intelligent messaging network servers;

wherein a transport protocol used within said intelligent messaging network provides for networking services comprising at least one of message segmentation and reassembly, message retries, message duplication detection, and message ACK and NACK service, said networking services being performed without relying on either a client application and server application.

48. (withdrawn) A (SDK), comprising:

a base registry key for storage of server settings and an user interface for viewing or editing the server settings;

screen based error logging; and

error components for writing system errors to an event log.

49. (withdrawn) An SDK, comprising:

utility components providing functions for a client application to run;

security components for encryption of messages; and

transport components for data communication over wireless networks, wherein the transport components are independent from both the security components and the utility components.

50.   (previously presented) The computer readable data storage medium of claim 1, wherein:

said second segment facilitates searching said database based on said server type to identify said second intelligent messaging network server, said second intelligent messaging network server being of a intelligent messaging network server type that said first intelligent messaging network server desires to connect with.

51.   (previously presented) The computer readable data storage medium of claim 50, wherein:

said second segment facilitates a handshake procedure determining a validity of a connection between said first intelligent messaging network server and said second intelligent messaging network server.

52.   (previously presented) The computer readable data storage medium of claim 1, wherein:

said intelligent messaging network server types are associated with functions performed by said plurality of intelligent messaging network servers.

53.   (previously presented) The computer readable data storage medium of claim 1, wherein:

said intelligent messaging network server types comprise at least one of a protocol gateway servers, message router servers, and back-end servers.

54. (previously presented) The computer readable data storage medium of claim 2, wherein:

said intelligent messaging network server class is associated with one of a network access protocol for a network connecting a client to said first intelligent messaging network server and an application executed by said first intelligent messaging network server.

55. (previously presented) The computer readable data storage medium of claim 1, wherein:

said third code segment encapsulates a network access protocol used to transmit data from a client device to said first intelligent messaging network server, wherein said network access protocol is transparent to said second intelligent messaging network server receiving said data from said first intelligent messaging network server.

56. (previously presented) The method of claim 24, wherein providing connectivity between the first intelligent messaging network server and the second intelligent messaging network server further comprises:

searching said database based on said server type to identify said second intelligent messaging network server, said second intelligent messaging network server being of an intelligent messaging network server type that said first intelligent messaging network server desires to connect with.

57. (previously presented) The method of claim 56, wherein providing connectivity between the first intelligent messaging network server and the second intelligent messaging network server further comprises:

facilitating a handshake procedure to determine a validity of a connection between said first intelligent messaging network server and said second intelligent messaging network server.

58. (previously presented) The method of claim 24, wherein:

said intelligent messaging network server types are associated with functions performed by said plurality of intelligent messaging network servers.

59. (previously presented) The method of claim 24, wherein the intelligent messaging network server types comprise:

at least one of a protocol gateway server, message router server, and back-end server.

60. (previously presented) The method of clam 25, wherein:

said intelligent messaging network server class is associated with at least one of a network access protocol for a network connecting a client to said intelligent messaging network first server and an application executed by said first intelligent messaging network server.

61. (previously presented) The method of claim 24, wherein encapsulating communication between the first intelligent messaging network server and the second intelligent messaging network server further comprises:

encapsulating a network access protocol used to transmit data from a client device to said first intelligent messaging network server, said network access protocol is transparent to said second intelligent messaging network server receiving said data from said first intelligent messaging network server.

62.   (currently amended) An apparatus comprising:

means for providing registration of a first intelligent messaging network server of a plurality of intelligent messaging network servers in an intelligent messaging network, wherein registration comprises storing a server id and a server type for said first intelligent messaging network server in a database storing server ids and server types for said plurality of intelligent messaging network servers;

means for providing connectivity of said first intelligent messaging network server to a second intelligent messaging network server of said plurality of intelligent messaging network servers; and

means for encapsulating communication between said first intelligent messaging network server and said second intelligent messaging network server;

wherein a transport protocol used within said intelligent messaging network provides for networking services comprising at least one of message segmentation and reassembly, message retries, message duplication detection, and message ACK and NACK service without relying on either a client application and server application.

63. (new) A method for supporting an intelligent messaging network servers in an intelligent messaging network, comprising:

providing an intelligent messaging network server in said intelligent messaging network;

using a transport protocol within said intelligent messaging network to facilitate communications between a client and a server;

wherein said transport protocol provides for networking services comprising at least one of message segmentation and reassembly, message retries, message duplication detection, and message ACK and NACK service, said networking services being performed by said intelligent messaging network server without relying on either a client and server.

64. (new) The method of claim 63, further comprising:
specifying a server class for said server.

65. (new) An intelligent messaging network, comprising:
an intelligent messaging network server in said intelligent messaging network;

a transport protocol within said intelligent messaging network to facilitate communications between a client and a server;

wherein said transport protocol provides for networking services comprising at least one of message segmentation and reassembly, message retries, message duplication detection, and message ACK and NACK service, said networking services being performed by said intelligent messaging network server without relying on either a client and server.

66. (new) The intelligent messaging network of claim 65, further comprising:

specifying a server class for said server.

67.   (new) An intelligent messaging network, comprising:

means for providing an intelligent messaging network server in said intelligent messaging network;

means for using a transport protocol within said intelligent messaging network to facilitate communications between a client and a server;

wherein said transport protocol provides for networking services comprising at least one of message segmentation and reassembly, message retries, message duplication detection, and message ACK and NACK service, said networking services being performed by said intelligent messaging network server without relying on either a client and server.

68.   (new) The intelligent messaging network of claim 67, further comprising:

means for specifying a server class for said server.